



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|-------------|----------------------|---------------------|------------------|
| 09/833,856 | 04/11/2001 | Vladimir Matena | SUNMP004 | 2226 |
| 25920 | 7590 | 12/30/2003 | EXAMINER | |
| MARTINE & PENILLA, LLP 710 LAKEWAY DRIVE SUITE 170 SUNNYVALE, CA 94085 | | | IQBAL, NADEEM | |
| | | ART UNIT | | PAPER NUMBER |
| | | 2114 | | 4 |
| DATE MAILED: 12/30/2003 | | | | |

Please find below and/or attached an Office communication concerning this application or proceeding.

| | | |
|------------------------------|------------------------|---------------------|
| Office Action Summary | Application No. | Applicant(s) |
| | 09/833,856 | MATENA ET AL. |
| | Examiner | Art Unit |
| | Nadeem Iqbal | 2184 |

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 11 April 2001.

2a) This action is FINAL. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-22 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-22 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

13) Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.
a) The translation of the foreign language provisional application has been received.

14) Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

| | |
|---|--|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s). _____ . |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449) Paper No(s) 2 . | 6) <input type="checkbox"/> Other: _____ . |

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1-6 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chapman et al., (U.S. Patent number 6594774) in view of Pagurek et al., (U.S. Patent number 6550024).

3. As per claim 1, Chapman teaches (col. 1, lines 58-60) an invention that provides reliability of a computer system by carrying out various monitoring functions, objects can register with a critical process monitor for various types of monitoring, also teaches failure detection to detect a failure of the object, and object providing recovery action upon detection of a failure of the object. He thus teaches limitations pertain to performing failure recovery, reporting a detection of an error in a system component to a runtime executive, notifying control module of the system component failure. He does not explicitly disclose isolating the system component upon detection of an error. He teaches (col. 12, lines 12-16) a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread. Pagurek et al., teaches (col. 2, lines 65-67) a system of locating errors and informing the appropriate agents so that they can avoid using the faulty resources in the future. He thus teaches to isolate the system components. It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the system of Pagurek et

al., that detects and isolates failed system components into the system of Chapman et al., to be able to detect errors and also able to isolate the failed system components. This is because Chapman already teaches as stated above a system that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, thus provides motivation for the stated inclusion.

4. As per claim 2, Chapman et al., already teaches as stated per claim 1 above a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, therefore the checkup thread clearly performs equivalent function of a service module.

5. As per claim 3, Chapman et al., already teaches determining a failed thread and setting a recovery event that identifies the failed thread, and Pagurek et al., teaches a system of locating errors and informing the appropriate agents so that they can avoid using the faulty resources in the future, therefore the combination as indicated per claim 1 clearly would provide disabling the service module.

6. As per claim 4, Chapman et al., teaches (col. 2, lines 40-42) an application programming interface that provides the interface between the monitor and the objects in the computer system, allowing objects to access the various features of the monitor, thus provides applicability to Java server.

7. As per claim 5, He already teaches per claim 3, that Pagurek teaches a system that locates errors, informs the appropriate agents so they can avoid using the faulty resources, therefore execution in a Java environment would enable stopping execution of a Java server.

8. As per claim 6, Chapman et al., already teaches determining a failed thread and setting a recovery event that identifies the failed thread, and Pagurek et al., teaches a system of locating errors and informing the appropriate agents so that they can avoid using the faulty resources in the future, therefore the combination as indicated per claim 1 clearly would provide a system that facilitate isolating a system component.

9. Claims 7-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chapman et al., (U.S. Patent number 6594774) in view of Bradshaw et al., (U.S. Patent number 6480944).

10. As per claim 7, Chapman substantially teaches the claimed invention as disclosed related to claim 1 above. He also teaches (col. 1, lines 58-60) an invention that carries out various monitoring functions, objects can register with a critical process monitor for various types of monitoring, also teaches failure detection to detect a failure of the object, and object providing recovery action upon detection of a failure of the object. He thus teaches limitations pertain to performing failure recovery, detecting an error, determining modules that are affected by the detected error. He does not explicitly disclose restarting the modules affected by the detected error on a second Java server using the control module. He teaches (col. 12, lines 12-16) a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread. Bradshaw et al., teaches (col. 13, lines 8-12) recovery strips and restart strips that include the ability to restart a process, the restart scripts also include the ability to restart a system failure. It would have been obvious to a person of ordinary skill in the art to include the recovery method of Bradshaw that include recovery strips and restart strips as taught by Bradshaw into the method of Chapman thus clearly provides the ability to restart the

modules affected by the detected error. This is because Chapman already teaches as stated above a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, Bradshaw further teaches that (abstract) stated features may reduce the amount of data and process time lost as a result of system failure, thus provides motivation for the inclusion.

11. As per claim 8, Bradshaw et al., also teaches (col. 13, lines 20-22) recovery strips also include the ability to recover lost data after failures and recovery strips may also enable a system to recover the overall operation of an application process or system, thus provides the ability to allocate a new module for an affected module as claimed.

12. As per claims 9 & 10, Bradshaw already teaches as stated above recovery strips also include the ability to recover lost data after failures and recovery strips may also enable a system to recover the overall operation of an application process or system, thus would clearly include operation of loading a state for the new module.

13. As per claims 11 & 12, Bradshaw also teaches (col. 13, lines 10-12) that the recovery strips also include the resetting or otherwise reconfiguration of an application so that it may resume operation, which would clearly require enabling routing of incoming messages to the new module and disabling routing of incoming messages to the module affected by the error.

14. As per claim 13, Bradshaw also teaches as stated above that the recovery strips also include the resetting or otherwise reconfiguration of an application, thus providing the ability to stop execution of the module affected by the error.

15. As per claim 14, Chapman substantially teaches the claimed invention as disclosed related to claim 1 above He also teaches (col. 1, lines 58-60) various monitoring functions,

objects can register with a critical process monitor for various types of monitoring, also teaches failure detection to detect a failure of the object, and object providing recovery action upon detection of a failure of the object. He thus teaches limitations pertain to performing failure recovery, reporting a detection of an error in a system component to a runtime executive, notifying control module of the system component failure. He does not explicitly disclose isolating the system component upon detection of an error. He teaches (col. 12, lines 12-16) a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread. Pagurek et al., teaches (col. 2, lines 65-67) a system of locating errors and informing the appropriate agents so that they can avoid using the faulty resources in the future. He thus teaches to isolate the system components. It would have been obvious to a person of ordinary skill in the art to include the system of Pagurek et al., that detects and isolates failed system components into the system of Chapman et al., to be able to detect errors and also able to isolate the failed system components. This is because Chapman already teaches as stated above a system that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, thus provides motivation for the stated inclusion.

16. As per claim 15, Chapman et al., already teaches as stated per claim 1 above a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, therefore the checkup thread clearly provides the ability to notify the control module of the component failure.

17. As per claim 16, Chapman et al., already teaches determining a failed thread and setting a recovery event that identifies the failed thread, and Pagurek et al., teaches a system of locating

errors and informing the appropriate agents so that they can avoid using the faulty resources in the future, therefore the combination as indicated per claim 1 clearly provides the ability to determine modules affected by the detected error.

18. As per claim 17, Chapman et al., does not explicitly disclose restarting the modules affected by the detected error. He teaches (col. 12, lines 12-16) a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread. Bradshaw et al., teaches (col. 13, lines 8-12) recovery strips and restart strips that include the ability to restart a process, the restart scripts also include the ability to restart a system failure. It would have been obvious to a person of ordinary skill in the art to include the recovery method of Bradshaw that include recovery strips and restart strips as taught by Bradshaw into the method of Chapman thus clearly provides the ability to restart the modules affected by the detected error. This is because Chapman already teaches as stated above a checkup thread that determines whether the selected thread has failed and if so sets a recovery event that identifies the failed thread, Bradshaw further teaches that (abstract) stated features may reduce the amount of data and process time lost as a result of system failure, thus provides motivation for the inclusion.

19. As per claims 18 & 19, Bradshaw et al., also teaches (col. 13, lines 20-22) recovery strips also include the ability to recover lost data after failures and recovery strips may also enable a system to recover the overall operation of an application process or system, thus provides the ability to allocate a new module for an affected module as claimed.

20. As per claims 20 & 21, Bradshaw already teaches as stated above recovery strips also include the ability to recover lost data after failures and recovery strips may also enable a system

to recover the overall operation of an application process or system, thus would clearly include operation of loading a state for the new module.

21. As per claim 22, Bradshaw also teaches (col. 13, lines 10-12) that the recovery strips also include the resetting or otherwise reconfiguration of an application so that it may resume operation, and also teaches that the recovery strips also include the ability to recover lost data after failures and recovery strips may also enable a system to recover the overall operation of an application process or system, thus the loaded state of a module would be a state of a module affected by the detected error prior to a time of detected error since he teaches recovery and reconfiguration such that the application may resume operation, which require loading a state prior to the detected error state.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nadeem Iqbal whose telephone number is (703)-308-5228. The examiner can normally be reached on M-F (8:00-5:30) First Friday Off.

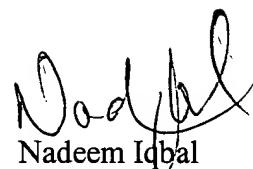
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert W Beausoliel can be reached on (703)-305-9713. The fax phone numbers for the organization where this application or proceeding is assigned are (703)-746-7239 for regular communications and (703)-746-7238 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703)-305-3900.

Application/Control Number: 09/833,856

Art Unit: 2184

Page 9


Nadeem Iqbal
Primary Examiner
Art Unit 2184